Facial Expression Recognition Based on MILBoost

Shaoping Zhu
Department of Information Management, Hunan University of Finance and Economics, 410205, China
E-mail: zhushaoping_cz@163.com

Abstract— **In this paper, We use Adaboost to create MILBoost and propose a new MILBoost approach to automatically recognize the facial expression from video sequences by constructing the MILBoost methods. At first, we determine facial velocity information using optical flow technique, which is used to charaterize facial expression. Then visual words based on facial velocity is used to represent facial expression using Bag of Words. Final MILBoost model is used for facial expression recognition, in order to improve the recognition accuracy, the class label information was used for the learning of the MILBoost model. Experiments were performed on a facial expression dataset built by ourselves and evaluated the proposed method, the experiment results show that the average recognition accuracy is over 89.2%, which validates its effectiveness.**

Index Terms—**Facial expression recognition; Motion feature; Bag of Words; MILBoost**

I. Introduction

In recent years, vision-based facial expression recognition is of great scientific and practical importance. Facial expression recognition has become a hot spot. Tremendous amount of researches have been carried out in the field of automatic facial expressions recognition from video sequence. Facial expression delivers rich information about human emotion and plays an important role in human communications. There are numerous significant theoretic values and wide potential applications for facial expressions recognition. It is a typical pattern analysis and recognition for facial expressions recognition as a scientific issue, which is understanding and classification problem, closely related to many disciplines such as Pattern Recognition, Computer Vision, Intelligent Human Computer Interaction, Computer applications, Graphics, and Cognitive Psychology etc. Facial expressions recognition is widely applied to public security, law enforcement, information security, financial security, patient care and cost reduction, such as mug shots retrieval, real-time video surveillance, bank cryptography and so on.

Facial expression recognition is an extensive and challenging research problem. Recently, It has made significant progresses in the technology of the facial expression recognition. Ekman and Friesen [1] began with scientific study of facial expressions. They analyzed six facial expressions, which included “surprise”, “fear”, “disgust”, “anger”, “happiness”, and “sadness”. Each expression was summarized using distinctive clues in the appearance of the eyebrows, eyes, mouth, jaw, etc, which were further investigated and encoded into the so called Facial Action Coding System (FACS) to describe “all visually distinguishable facial movements.” Many approaches were presented to recognize facial expressions by machines. These methods can be categorized for four types as follow: In the first type, emotion space was used to recognize facial expression [2]. For the second type, it is to recognize facial expressions of an image frame sequence by the use of optical flow [3]. The third type is to recognize facial expressions by active shape models [4]. The fourth type is to use neural network namely to recognize facial expressions, which exploit neural models to capture and encode the nonlinear mapping among different facial expressions [5]. However, these approaches have been fraught with difficulty because they are often inconsistent with other evidence of facial expressions [6].

It is essential for intelligent and natural human computer interaction to recognize facial expression automatically. In the past several years, significant efforts have been made to identify reliable and valid facial indicators of expressions [7-19]. In [7-9], an approach was developed to automatically recognize facial expressions. Active Appearance Models (AAM) was used to decouple shape and appearance parameters from face images. And then SVM were used to classify facial expressions. In [10-14], Prkachin and Solomon validated a Facial Action Coding System (FACS) based measure of pain that could be applied on a frame-by-frame basis. But these methods require manual labeling of facial action units or other observational measurements by highly trained observers [20, 21], which is both timely and costly. Most must be performed offline, which makes them ill-suited for real-time applications. In [15], a robust approach for expression recognition was presented using

Manuscript received January 10, 2014; revised February, 2014; accepted March, 2014

Foundation item: Project supported by Research Foundation of Hunan Provincial Science & Technology Department under Grant No.2014FJ3057, by the Teaching Reform Foundation of Hunan Province Ordinary College under Grant (NO. 2012401544), and by the Foundation for Key Constructive Discipline of Hunan Province.

Corresponding author: Shaoping Zhu, Associate Professor.

doi:10.4304/jsw.9.9.2435-2442

video sequences. An automatic face detector is employed which uses skin color modeling to detect human face in the video sequence. Facial expressions are obtained by using a mask image. The obtained face images are then projected onto a feature space, defined by Eigenfaces, to produce the biometric template. Facial expressions recognition is performed by projecting a new image onto the feature spaces spanned by the Eigenfaces and then classifying the facial expressions by comparing its position in the feature spaces with the positions of known individuals. Zhang [16] used supervised locality preserving projections (SLPP) to extract feature of expression, and multiple kernels support vector machines (MKSVM) is employed for recognizing expression. Methods described above use static features to characterize facial expression, but these static features cannot fully represent facial expressions. Juanjuan C, et. al [22] proposed a way of facial expression recognition by reconstructing the PCA. They separated the training set into seven parts by expression, and got the Orthonormal Basis of each subset expression by PCA algorithm, then marked projection with the original image on each of the Orthonormal Basis, rebuilt the projection coordinate and recognized the expression by the combination of D-value summation and similarity methods.

However, evaluation results and practical experience have shown that facial expression automatically technologies are currently far from mature. Many challenges are to be solved before it can implement a robust practical application. In this paper, we propose a method for automatically recognizing facial expressions from video sequences. This approach includes two steps: extracting feature of facial expressions and classifying facial expressions. In the extracting feature, features of facial expressions are extracted by motion descriptor based on optical flow. Then we convert facial velocity information to visual words using "bag-of-words" models, and facial expression is represented by a number of visual words; Final the multiple instance boosting (MILBoost) model is used for facial expression recognition. In addition, in order to improve the recognition accuracy, the class label information is used for the learning of the MILBoost model.

The paper is structured as follows. After reviewing related work in this section, we describe the facial expression feature extraction base on optical flow technique and "bag-of-words" models in section 2. Section 3 gives details of MILBoost model for recognize facial expression. Section 5 shows experiment result, also comparing our approach with three state-of-the-art methods, and the conclusions are given in the final section.

II. Facial Expression Representation

A Facial Velocity Feature

Efficient face representation is the key for a good facial expression recognition method. For its eminent characteristics in spatial local feature exaction and orientation selection, Optical flow-based face representation has attracted much attention. According to the physiology, the expression is a dynamic event, it must be represented by the motion information of the face. So, we use facial velocity features to characterize facial expression. The facial velocity features (optical flow vector) are estimated by optical flow model, and each facial expression is coded on a six level intensity dimension (A–F): "surprise", "fear", "disgust", "anger", "happiness", and "sadness".

Given a stabilized video sequence in which the face of a person appears in the center of the field of view, we compute the facial velocity (optical flow vector) ***u***=(u_x ,u_y) at each frame using optical flow equation, which is expressed as :

$$I_x u_x + I_y u_y + I_t = 0 \tag{1}$$

Where

$$I_x = \frac{\partial I}{\partial x}, \qquad I_y = \frac{\partial I}{\partial y}, \qquad I_t = \frac{\partial I}{\partial t}$$

$$u_x = \frac{dx}{dt}, \qquad u_y = \frac{dy}{dt} \quad ,$$

where (x, y, t) is the image in pixel (*x*, *y*) at time *t*, where $I(x, y, t)$ is the intensity at pixel (*x*, *y*) at time *t*, u_x ,u_y is the horizontal and vertical velocities in pixel (*x*, *y*).

We can obtain ***u***=(u_x ,u_y) by minimizing the objective function:

$$C = \int_D \left[\lambda^2 \|\nabla u\|^2 + (\nabla I \cdot u + I_t)^2 \right] dxdy \tag{2}$$

There are many methods to solve the optical flow equation. We use the iterative algorithm [23] to compute the optical flow velocity：

$$\begin{aligned} u_x^{k+1} &= \bar{u}_x^k - \frac{I_x\left[I_x \bar{u}_x^k + I_y \bar{u}_y^k + I_t\right]}{\lambda + I_x^2 + I_y^2} \\ u_y^{k+1} &= \bar{u}_y^k - \frac{I_y\left[I_x \bar{u}_x^k + I_y \bar{u}_y^k + I_t\right]}{\lambda + I_x^2 + I_y^2} \end{aligned} \qquad （3）$$

Where k is the number of iterations, initial value of velocity $u_x^0 = u_y^0 = 0$, $\bar{u}_x^k, \bar{u}_y^k$ is the average velocity of the neighborhood of point (*x*,*y*).

The optical flow vector field ***u*** is then split into two scalar fields u_x and u_y, corresponding to the *x* and *y* components of ***u*** [24]. u_x and u_y are further half-wave rectified into four none- gative channels u_x^+, u_x^-, u_y^+,u_y^-, so that u_x= u_x^+-u_x^- and u_x= u_x^+-u_x^-. These four nonnegative channels are then blurred with a Gaussian kernel and normalized to obtain the final four channels ub_x^+, ub_x^-, ub_y^+,ub_y^-.

Facial expression is represented by velocity features that are composed of the channels ub_x^+, ub_x^-, ub_y^+,ub_y^- of all pixels in facial image. Because facial expression can be regard as facial motion, the velocity features can describe facial expression effectively, in addition to, the velocity features have been shown to perform reliably with noisy image sequences, and has been applied in various tasks, such as action classification, motion synthesis, etc. But the dimension of these velocity features is too high (6 x $L \times L$, where $L \times L$ is image size) to be used directly to recognition, so, we convert these velocity features into visual words using "bag-of- words" [25, 26].

B. *Visual Words for Characterizing Facial Expression*

The "bag-of-words" model was originally proposed for analyzing text documents, where a document is represented as a histogram over word counts.

In this paper, each facial image is divide into $L\, xL$ blocks , and each image block is represented by optical flow vector of all pixels in the block. On this basis，Facial expressions are represented by visual words using the method of BoW (Bag of Words) .

To construct the codebook, we randomly select a subset from all image blocks, then, we use k-means clustering algorithms to obtain clusters. Codewords are then defined as the centers of the obtained clusters, namely visual words. In the end, each face image is converted to the "bag-of-words" representation by appearance times of each codeword in the image is used to represent the image, namely BoW histogram.

The step for characterizing facial expression is as follows：

Step1: Optical flow channels ub_x^+, ub_x^-, ub_y^+,ub_y^- are computed；

Step2: Each facial image is divide into $L \times L$ blocks, which is represented by optical flow vector of all pixels in the block；

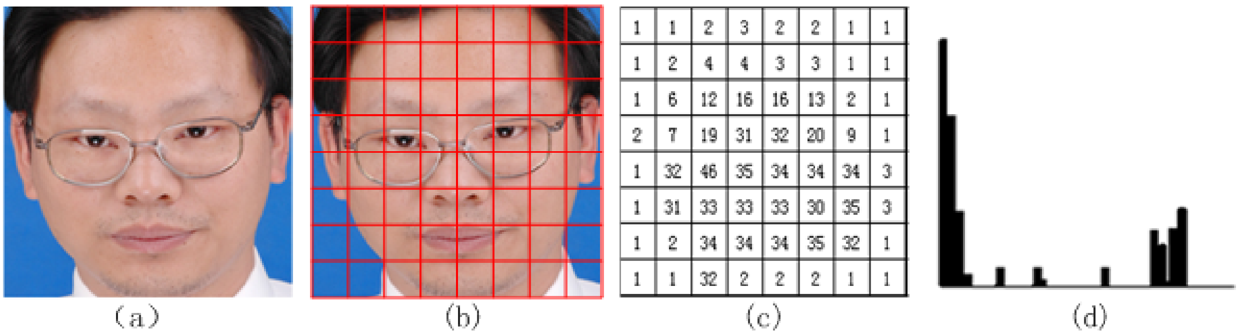
Step3: Vision words are obtained using *k*-means clustering algorithms;

Step4: Facial expression is represented by BoW histogram x_{ij}:

$$x_{ij} = \{n(I,d_1),\cdots,n(I,d_j),\cdots,n(I,d_M)\} \qquad (4)$$

where， $n(I,d_j)$ is the number of visual word d_j included in image, M is the number of vision words in word sets.

Fig. 1 shows an example of our "bag-of-words" representation.



1	1	2	3	2	2	1	1
1	2	4	4	3	3	1	1
1	6	12	16	16	13	2	1
2	7	19	31	32	20	9	1
1	32	46	35	34	34	34	3
1	31	33	33	33	30	35	3
1	2	34	34	34	35	32	1
1	1	32	2	2	2	1	1

Fig. 1. The processing pipeline of the "bag-of-words" representation: (a) given a image,(b) divide into $L \times L$ blocks (c) represent each block by a "visual word", and (d) ignore the ordering of words and represent the facial image as a histogram over "visual words".

III. MILBoost-Based Facial Expression Recognition

Facial expression recognition is innately a Multiple Instance Learning problem. We use the multiple instance boosting(MILBoost) models to learn and recognize facial expression. Our approach is directly inspired by a body of work on using generative MILBoost models for visual recognition based on the "bag-of-words" paradigm. The MILBoost models have been applied to various computer vision applications, such as object recognition, action recognition, human detection, etc. We use Adaboost [27] to create MILBoost and propose a new MILBoost framework, which learns a unified classifier instead of individual classifiers for all classes, so that the recognition efficiency can be increased without compromising accuracy.

A. *Adaboost Algorithm Analysis*

Freund and Schapire proposed Adaboost algorithm in 1995 [28]. Adaboost is one of the most efficient machine learning algorithms at present.

Suppose we give a set of training data $S = \{(x_1, y_1),(x_2, y_2),...,(x_l, y_l)\}$, where x_i denotes the *i*-th image blocks, $y_i \in \{0,1\}$, "1" indicates positive example samples and "0" indicates negative example samples, and (x_i, y_i) denotes the sample of the *i*-th image blocks. $1 \le i \le l$, l is the number of samples. The AdaBoost algorithm is an iterative procedure, which tries to approximate the Bayes classifier by combining many weak classifiers. At first the AdaBoost builds a classifier at the begin with the unweighted training sample, which produces class labels for sample a classification tree .Then, the weight of that training data point is increased if a training data point is misclassified. A second classifier is built by using the new weights for no longer equal. Again, misclassified training data have their weights boosted and the procedure is repeated. Finally a score is assigned to each classifier, and the final classifier is defined as the linear combination of the classifiers from each stage. The AdaBoost algorithm proceeds as follows:

Give: $S = \{(x_1, y_1),(x_2, y_2),...,(x_l, y_l)\}$, where $x_i \in S$, $y_i \in \{0,1\}$

Step 1: Initialize the observation weights: $w_{1,i}=\frac{1}{2l_0}$, $\frac{1}{2l_1}$, where $w_{1,i}$ is corresponding to a weight of the *i*-th sample at the first time, l_0 is the total number of positive example samples, l_1 is the total number of negative example samples.

Step 2: The training sample set is trained for *T* rounds of training.

For *t* = 1 to *T* do

Train weak learner using distribution D_t, which fit a classifier $h_j(x_i)$ to the training data using weights w_t. Get weak hypothesis $h_t(\cdot)=h_k(\cdot)$, where $\forall j\neq k, \varepsilon_k<\varepsilon_j$, and $\varepsilon_t=\varepsilon_k$

Select h_t with low weighted error. We assume that ε_j is classification error rate and calculate: $\varepsilon_j=\Pr_i^{w_t}[h_j(x_i)\neq y_i]$, which w_t is corresponding to a weight of the sample at the *t*-th time, *j* represents each feature, $1\leq j\leq n$. *n* is the total number of alternative features, h_j is a weak classifier of feature *j*, we define:

$$h_j(x)=\begin{cases}1, & p_j f_j(x)<p_j\theta_j\\ 0, & \text{Others}\end{cases} \tag{5}$$

where $f_j(x)$ is eigenvalue, θ_j is threshold, and p_j is a symbol factor of the inequality direction.

Update the weight of the sample by choosing β_t:

$$w_{t+1,i}=w_{t,i}\beta_t^{1-\varepsilon_i} \tag{6}$$

where $\varepsilon_i=\{0,1\}$, $\beta_t=\frac{1}{2}In\frac{\varepsilon_t}{1-\varepsilon_t}$.

Step3: Output the final hypothesis:

$$h(x)=\sum_{t=1}^{T}a_t h_t(x)\geq\frac{1}{2}\sum_{t=1}^{T}a_t \tag{7}$$

where $a_t=\log\frac{1}{\beta_t}$.

AdaBoost has been proved to be extremely successful in producing accurate classifiers. However, it is much harder to achieve in the multi-class case, which was also proposed to be used in the multi-class case. The main disadvantage of AdaBoost is that it is unable to handle weak learners with an error rate greater than 1/2. However, AdaBoost may easily fail in the multi-class case. We have combined MIL with AdaBoost to create MILBoost, where that the bag is positive can produce classifiers with much higher recognition rates and fast computation times.

B. MILBoost-based Facial Expression Recognition

Keeler, et. al [29] proposed originally the idea for the multiple instance learning for handwritten digit recognition in 1990. It was called Integrated Segmentation and Recognition (ISR), and it is the key idea to provide a different way in constituting training samples. Training samples are not singletons, at the same time they are in "bags", where all of the samples in a bag share a label [30]; Samples are organized into positive bags of instances and negative bags of instances, where each bag may contain a number of instances [31]. At least one instance is positive (i.e. object) in a positive bag, while all instances are negative (i.e. non-object) in a negative bag. In MILBoost, learning must simultaneously learn that samples in the positive bags are positive along with the parameters of the classifier. To obtain training samples, each image is divided into $L\,xL$ blocks. We treat each block in a image as a single word d_j and a image as a bag. Each block is used as an example for the purposes of training. It is suitable to represent the object by a bag of multiple instances (non-aligned human face images). Then, MILBoost can learn that instances in the positive bags are positive, along with a binary classifier [32]. In this paper, MILBoost is employed for facial expression with non-aligned training samples. The MILBoost-based facial expression recognition proceeds as follows:

Step 1: Input: Given dataset $\{X_i,y_i\}_{i=1}^{N}$, X_i is training bags, where $X_i=\{x_{i1},x_{i2},\cdots,x_{ij},\cdots,x_{iN}\}$, y_i is the score of the sample, and $y_i\in\{0,1\}$. *N* is the number of all weak classifiers. A positive bag contains at least one positive sample, and. $y_i=\max_j(y_{ij})$

Pick out *K* weak classifiers and consist of strong classifier.

Step 2: Update all *N* weak classifiers in the pool with data $\{x_{ij},y_i\}$.

Step 3: Initialize all strong classifier: $H_{ij}=0$ for all *i*, *j*.

Step 4: for *k*=1 to *K* do

for *m*=1 to *N* do

We calculate the probability that the j-th sample is positive in the i-th bag as follow:

$$P_{ij}^{m}=\sigma(H_{ij}+h_m(x_{ij})) \tag{8}$$

Where $P_{ij}^{m}=p(y_i|x_{ij})=\frac{1}{1+\exp(-y_{ij})}$.

We calculate the probability that the bag is positive as follow:

$$P_i^m=1-\prod\nolimits_j(1-p_{ij}^m) \tag{9}$$

Where $P_i^m=\mathrm{p}(y_i\mid X_i)$.

The likelihood assigned to a set of training bags is:

$$C^m=\sum\nolimits_i(y_i\log(p_i^m)+(1-y_i)\log(1-(p_i^m)) \tag{10}$$

End for

Finding the maximum *m** from *N* as the current optimal weak classifier as follow:

$$m^*=\arg\min\nolimits_m C^m \tag{11}$$

The *m** come into the strong classifier:

$$h_k(x)\leftarrow h_{m*}(x) \tag{12}$$

$$H_{ij}=H_{ij}+h_k(x) \tag{13}$$

End for

Step 5: Output: Strong classifier which consist of K weak classifiers as follow:

$$H(x) = \sum_k h_k(x) \tag{14}$$

where h_k is a weak classifier and can make binary predictions using $sign(H_K(x))$.

In MILBoost, samples come into positive bags of instances and negative bags of instances. Each instance x_{ij} is indexed with two indices, where i for the bag and j for the instance within the bag. All instances in a bag share a bag label y_i. Weight of each sample composes of the weight of the bag and the weight of the sample in the bag. The quantity of the samples can be interpreted as a likelihood ratio, where some (at least one) instance is positive in a bag. P_{ij}^m is the probability that some instance is positive. So the weight of samples in the bags is P_{ij}^m. We calculate: $w_{ij} = \frac{\partial \log C^m}{\partial y_{ij}}$, and get weight of the bags w_{ij}.

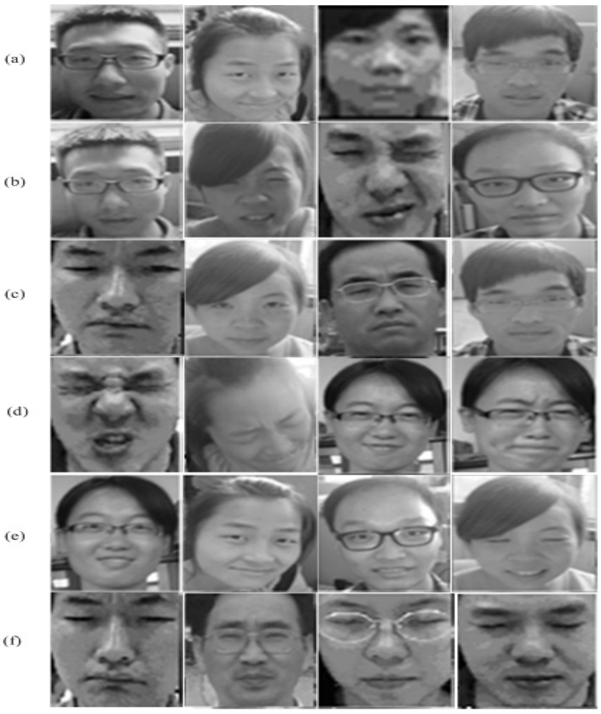
Training in the initial stages is the key to a fast and effective classifier. Training and evaluating have a direct impact on both the features selected and the appropriate thresholds selected. The result of the MILBoost learning process is not only a sample classifier but also weights of the samples. The samples have high score in positive bags which are assigned high weight. The final classifier labels these samples to be positive. The remaining samples have a low score in the positive bags, which are assigned a low weight. The final classifier classifies these samples as negative samples as they should be. We train a complete MILBoost classifier and set the detection threshold to achieve the desired false positive rates and false negative rates. Retrain the initial weak classifier so that a zero false negative rate is obtained on the samples, which label positive by the full classifier. This results in a significant increase in many samples to be pruned by the classifier. Repeating the process so that the second classifier is trained to yield a zero false negative rate on the remaining samples.

For the task of facial expression recognition classification, our goal is to classify a new face image to a specific facial expression class. During the inference stage, given a testing face image, We can treat each aspect in the MILBoost model as one class of facial expression. For facial expression recognition with large amount of training data, this will result in long training time. In this paper, we adopt a supervised Algorithm to train MILBoost model. The supervised training algorithm not only makes the training more efficient, but also improves the overall recognition accuracy significantly. Each image has a class label information in the training images, which is important for the classification task. Here, we make use of this class label information in the training images for the learning of the MILBoost model, since each image directly corresponds to a certain facial expression class on train sets.

IV. Experimental Results and Analysis

We verified the effectiveness of the proposed algorithm by using C++ and Matlab hybrid implementation on a PC with Haswell 3.2 GHz processor and 4G RAM.

We have built a database of face images about six facial expressions. In this database, there are six groups of images, which are “surprise”, “fear”, “disgust”, “anger”, “happiness”, and “sadness”, and each group includes 15 males and 15 females. The face images were taken under various laboratory controlled lighting conditions, and each face image was normalized to a size of 64×64. Some sample images are shown in Fig.2



(a) surprise,(b) fear,(c) disgust,(d) anger, (e) happiness,(f) sadness
Fig.2 Examples of recognizing pain from facial videos

In Experiments, we chose 25 face images per class randomly for training, while the remaining images were used for testing. These images were pre-processed by aligning and scaling so that the distances between the eyes were the same for all images and also ensuring that the eyes occurred in the same coordinates of the image. We run the system six times and obtained six different training and testing sample sets. The recognition rates were found by averaging the recognition rate of each run.

Each face image was divided into $L \times L$ blocks. First, we studied the effect of the size of image block on the recognition accuracy. Figure 3 represent the recognition accuracy curve with different block sizes L. It can be concluded that the accuracy peaked when the block sizes L is 8. Therefore L is set as 8.

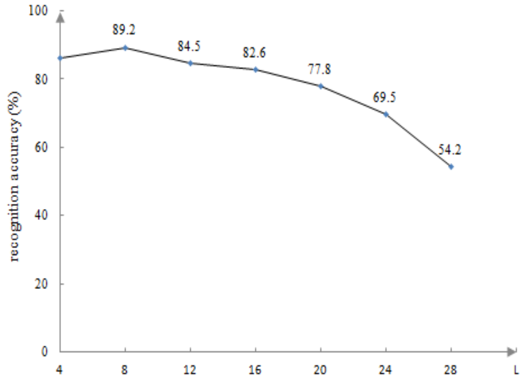


Fig.3 Recognition accuracy curve with different block sizes

In order to determine the value of *M* that is the number of the visual word set, the relation between *M* and recognition accuracy was observed, which is displayed in Fig.4. It is revealed in Fig.4 that with the increasing of *M* recognition accuracy is rise up at the beginning and if *M* is larger than or equal to 50, the recognition accuracy is stabled to 0.892. As a result, *M* is set as 50.

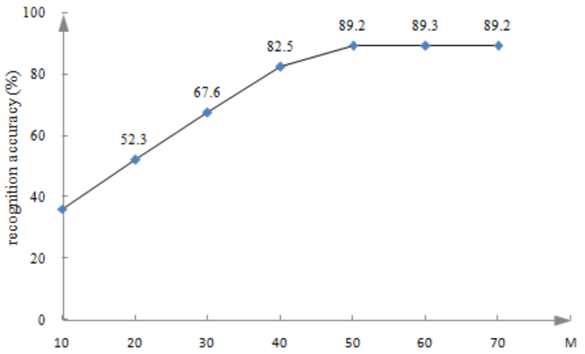


Fig.4 Relation curve between M and accuracy

To examine the accuracy of our proposed facial expressions approach, 300 different face images were used for this experiment. Some images contained the same person but in different expressions. The recognition results are presented in the confusion matrices. The confusion matrix for per-video classification is shown in Figure 6, which built a database of face images using 50 codewords. Where A,B,C,D,E,F indicate "surprise", "fear", "disgust", "anger", "happiness", and "sadness" respectively.

A	0.89	0.06	0.02	0.01	0	0.02
B	0.05	0.88	0.03	0	0.01	0.03
C	0.04	0.04	0.86	0	0.01	0.05
D	0.02	0.02	0.01	0.94	0	0.01
E	0.01	0.01	0.03	0	0.93	0.02
F	0.04	0.02	0.07	0.01	0.01	0.85
	A	B	C	D	E	F

Figure 5 Confusion matrix for expression recognition of our method

Each cell in the confusion matrix is the average result of every facial expression respectively. We can see that the algorithm correctly classifies most facial expressions. Average recognition rate get to 89.2%. Most of the mistakes the algorithm makes are confusions between "surprise" and "fear", between "disgust" and "sadness". This is intuitively reasonable since they are similar facial expressions.

In order to examine the accuracy of our proposed facial expression recognition approach, we compared our method to three state-of-the-art approaches for facial expression recognition using the same data. The first method is "AAM+SVM" [7], which used Active Appearance Models (AAM) to extract face features, and SVM to classify facial expression. The second method is "Eigenimage" [15], which used Eigenface for facial expression recognition. The third method is "SLPP+ MKSVM" [16], which used SLPP to extract feature of facial expression, and multiple kernels support vector machines (MKSVM) for recognizing. 200 different expression images were used for this experiment. Some images contained the same person but in different facial expression. The results of recognition accuracy comparison are shown in Figure 6.

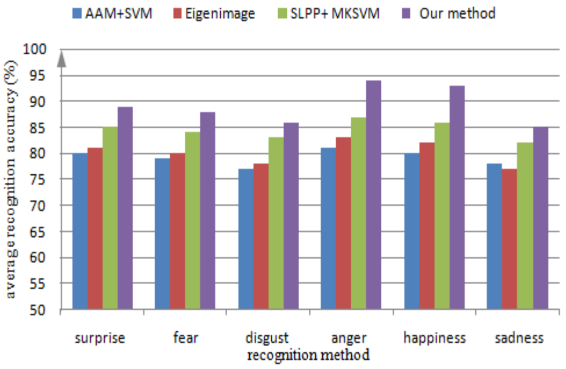


Fig.6 Recognition accuracy comparison of different method

In Figure 6, we show that AAM+SVM[7] obtain average recognition accuracy of 80.3%. The average recognition rate of "Eigenimage"[15] is to 81.6%. The average recognition accuracy of SLPP+ KSVM [16] is to 84.5%. Our method is stabled to average recognition accuracy of 89.2% .We can see that our method performs significantly better than the above three state-of-the-art approaches for facial expression recognition. The reason is that we improved the recognition accuracy in the two stages of facial expression feature extraction and facial expression recognition. In the stage of facial expression feature extraction, we used motion features that were reliably with noisy image sequences and bag-of-words framework to describe facial expression effectively. In the stage of expression recognition, we used MILBoost algorithm to classify expression images. Our method performs the best, its recognition accuracies and speeds are satisfactory.

V. Conclusion

Facial expression recognition can provide significant advantage in public security, law enforcement, information security, financial security, patient care and cost reduction. In this paper, we have presented a novel method to recognize the facial expression and given the six facial expression levels at the same time. The main contribution can be concluded as follows:

(1) Visual words are used for facial expression. Optical flow model is used for extracting facial velocity features, then we convert facial velocity features into visual words using "bag-of-words" models.

(2) We use MILBoost model for facial expression recognition. In our models, Adaboost is used to create MILBoost and a new MILBoost framework is proposed, which recognize different expression categories. In addition, in order to improve the recognition accuracy, the class label information is used for the learning of the MILBoost model.

(3) Experiments were performed on a facial expression dataset built by ourselves and evaluated the proposed method. Experimental results reveal that the proposed method performs better than previous ones.

Acknowledgments

This work was supported by Research Fundation for Science & Technology Office of Hunan Province under Grant (NO.2014FJ3057), by Hunan Provincial Education Science and "Twelve Five" planning issues (NO. XJK012CGD022), by the Teaching Reform Foundation of Hunan Province Ordinary College under Grant (NO. 2012401544) and by the Foundation for Key Constructive Discipline of Hunan Province.

Conflict of Interests

The author declares that there is no conflict of interests regarding the publication of this article.

References

[1] P. Ekman and V. F Wallace, *Manual for the Facial Action Coding System*, Consulting Psychologists Press, Palo Alto, 1978.

[2] S. Morishima and H. Harashima, "Emotion Space for Analysis and Synthesis of Facial Expression," *Proceedings of 2nd IEEE International Workshop on Robot and Human Communication, 1993*, pp. 188-193.

[3] M. Rosenblum, Y. Yacoob, and L. S. Davis, "Human expression recognition from motion using a radial basis function network architecture," *IEEE Transactions on Neural Networks*, vol.5, issue7, 1996, pp. 1121-1138.

[4] T. F. Cootes, C. J. Taylor, D. H. Cooper and J l. Graham, "Active shape models-their training and application," in *Computer vision and image understanding*,vol.1,issue 61, 1995, pp. 38-59.

[5] K. Matsuno, C. W. Lee, S. Kimura and S. Tsuji, "Automatic recognition of human facial expressions," In *Proceedings of 5th International Conference on Computer Vision,* pp. 352-359, June 1995.

[6] D C. Turk, C. Dennis and R. Melzack, "The measurement of pain and the assessment of people experiencing pain," in *Handbook of Pain Assessment,* D C Turk and R. Melzack, Eds. New York: Guilford, 2001, 2nd edition: pp. 1-11.

[7] A. B. Ashraf, S. Lucey, J. F. Cohn,T. Chen, Z. Ambadar, K. M. Prkachin, et al., "The Painful Face: pain expression recognition using active appearance models," in *Image and Vision Computing, the 9th ACM International Conference on Multimodal Interfaces*, Nagoya, Aichi, Japan: ACM, 2007, pp. 9-14.

[8] A B. Ashraf, S. Lucey, J. F. Cohn, K M. Prkachin, and P. Solomon, "The painful face II-pain expression recognition using active appearance models," in *Image Vision Computing,* vol. 12, issue 27, 2009, pp. 1788-1796.

[9] L. Wang, R. F. Li, and K. Wang, "A novel automatic facial expression recognition method based on AAM," in *Journal of Computers,* vol 9, issue 3, 2014, pp. 608-617.

[10] P. Lucey, J. F Cohn, K. M. Prkachin, P. E. Solomon, S. Chew and I. Matthews, "Painful monitoring: Automatic pain monitoring using the UNBC-McMaster shoulder pain expression archive database," in *Image and Vision Computing*, vol. 3, issue 30, 2012, pp. 197-205.

[11] P. Lucey, J. Cohn, S. Lucey, I. Matthews, S. Sridharan and K. M. Prkachin, "Automatically detecting pain using facial actions," *3rd International Conference on Affective Computing and Intelligent Interaction and Workshops,* pp. 1-8, September 2009.

[12] P. Lucey, J. F. Cohn, I. Matthews, S. Lucey, S. Sridharan, J. Howlett, et al. "Automatically detecting pain in video through facial action units," in *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics,* vol. 3, issue 41, 2011, pp. 664-674.

[13] K. M. Prkachin, "The consistency of facial expressions of pain: a comparison across modalities," in *Pain*, vol. 3, issue 51, 1992, pp. 297-306.

[14] K. M. Prkachin and P. E. Solomon, "The structure, reliability and validity of pain expression: Evidence from patients with shoulder pain," in *Pain*, vol. 2, issue 139, 2008, pp. 267-274.

[15] M. Monwar, S. Rezaei, and K. Prkachin, "Eigenimage based pain expression recognition," in *IAENG International Journal of Applied Mathematics,* vol. 2, issue 36, 2007, pp. 1-6.

[16] Zhang Wei and Xia Li-min, "Pain expression recognition based on SLPP and MKSVM," in *International Journal of Engineering and Manufacturing*, issue 3, 2011, pp. 69-74.

[17] K. M. Prkachin, "The consistency of facial expressions of pain: a comparison across modalities," in *Pain,* vol. 3, issue 51, 1992, pp. 297-306.

[18] K. M. Prkachin and S. R. Mercer, "Pain expression in patients with shoulder pathology: validity, properties and relationship to sickness impact," in *Pain*, vol. 3, issue 39, 1989, pp. 257-265.

[19] Y． Ouyang, and N. Sang, "Robust automatic facial expression detection method,", in *Journal of Software,* vol 8, issue 7, 2013, pp.1759-1764.

[20] J. F. Cohn, Z. Ambadar, and P. Ekman, "Observer-based measurement of facial expression with the facial action coding system ," in *The handbook of emotion elicitation and assessment,* New York, NY: Oxford University Press, 2007, pp. 203-221.

[21] P. Ekman, W. V. Friesen, and J. C. *Hager, Facial Action Coding System: Research Nexus*, Salt Lake City, UT: Network Research Information, 2002.

[22] C. Juanjuan, Z. Zheng, S. Han and Z. Gang, "Facial expression recognition based on PCA reconstruction," in *the 5th International Conference on IEEE Computer Science and Education (ICCSE)*, 2010, pp.195-198.

[23] B. K. Horn and B. G. Schunck, "Determining optical flow," in *Artificial Intelligence*， issue 17, 1981, pp. 185-204.

[24] A. A. Efros, A. C. Berg, G. Mori and J Malik, "Recognizing action at a distance," *the 9th IEEE International Conference on Computer Vision. Nice, France: IEEE,* issue 2, 2003, pp 726-733.

[25] Y. Zhang, R. Jin, and Z. H. Zhou, “Understanding bag-of-words model: a statistical framework,” in *International Journal of Machine Learning and Cybernetics*, vol. 1-4, issue 1, 2010, pp. 43-52.

[26] T. Li, T. Mei, I. S. Kweon and X. S. Hua, “Contextual bag-of-words for visual categorization,” the *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 4, issue 21, 2011, pp. 381-392.

[27] R. E.Schapire, E Robert, and Y. Singer, “Improved boosting algorithms using confidence-rated predictions,” in *Machine learning*, vol. 3, issue 37, 1999, pp. 297-336.

[28] Y. Freund and R. E. Schapire, “A decision theoretic generalization of on-line learning and an application to boosting,” in *Journal of Computer and System Sciences*, vol. 1, issue 55(1), pp. 119–139, August 1997.

[29] J. D. Keeler, D. E. Rumelhart, and W. K. Leow, “Integrated segmentation and recognition of hand-printed numerals,” in *NIPS-3: Proceedings of the 1990 conference on Advances in neural information processing systems 3*, San Francisco, CA, USA: Morgan Kaufmann Publishers Inc, 1990, pp. 557–563,

[30] T. G.Dietterich, R. H. Lathrop, and T. Lozano-Pérez, “Solving the multiple instance problem with axis-parallel rectangles,” in *Artificial Intelligence*, vol. 1, issue 89, 1997, pp. 31–71.

[31] O. Marson, and T. Lozano-Perez, “A framework for multiple instance learning,” Advances in *neural information processing systems,* 1998, pp. 570-576.

[32] B. Babenko, P. Dollár, Z. Tu, and S. Belongie, “Simultaneous learning and alignment: Multi-instance and multi-pose learning,” in *Workshop on Faces in 'Real-Life' Images: Detection, Alignment, and Recognition, 2008.*

Shaoping Zhu received the MSc degree from Central South University, China, in computer science. She is currently an associate Professor in Hunan University of Finance and Economic. Her research interests include high-level recognition problems in computer vision, in particular, human facial expression recognition, human activity recognition, object and scene recognition, etc. She is an auther of over 30 joural and conference papers.